



DISTRIBUTED PROGRAMMING

SOCKET-BASED CHATTING

SO-CHAT

Presented by 4KE

AGENDA

INTRODUCTION



Introducing our Chat Application project

PROJECT DESIGN & ARCHITECTURE



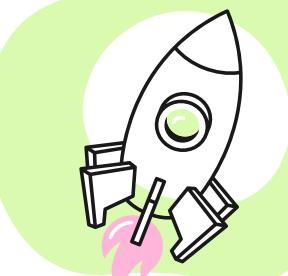
How our project is implemented

TESTING & RESULTS



Demonstrating how to run the program and its results

FUTURE WORK



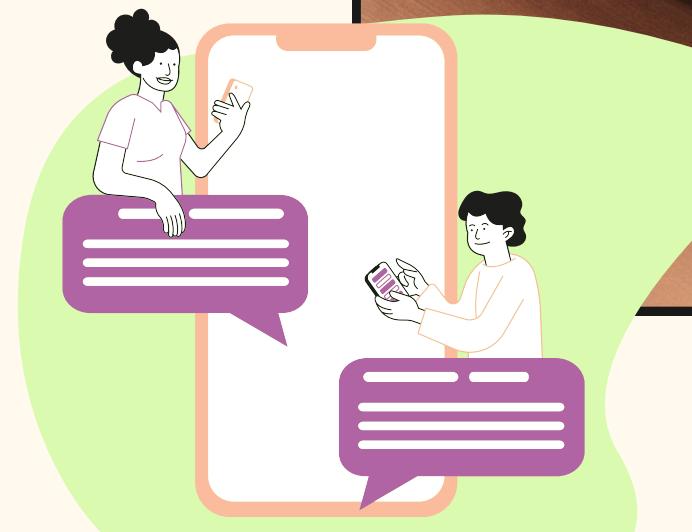
Our plans for Future Work



INTRODUCTION

INTRODUCTION

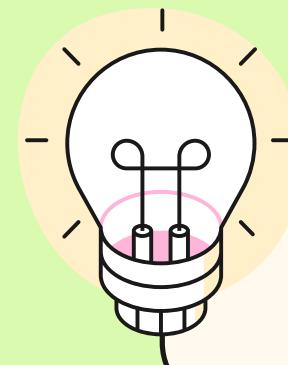
- In today's fast-paced world, the demand for instant, real-time communication is paramount.
- Whether for personal or professional purposes, the ability to connect and communicate seamlessly is a fundamental need.



OUR SOLUTION



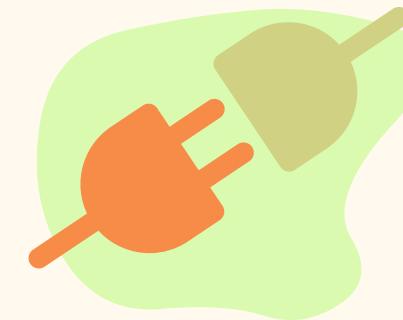
SO-CHAT



SOCKET-BASED CHATTING APP

- A user-friendly and feature-rich Chat Application
- Enables individuals and groups to exchange messages in real-time over a network

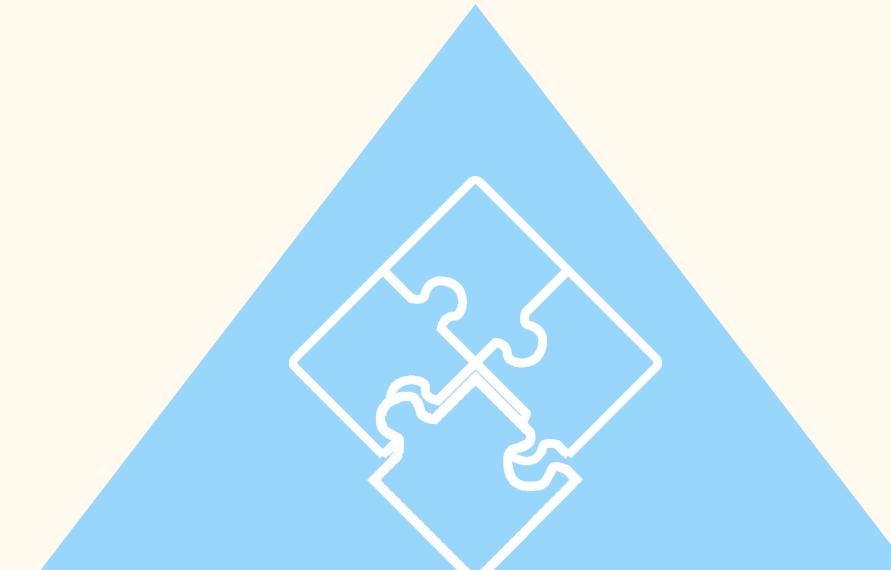
3 - PILLARS IN OUR SOLUTION



Socket-
Programming

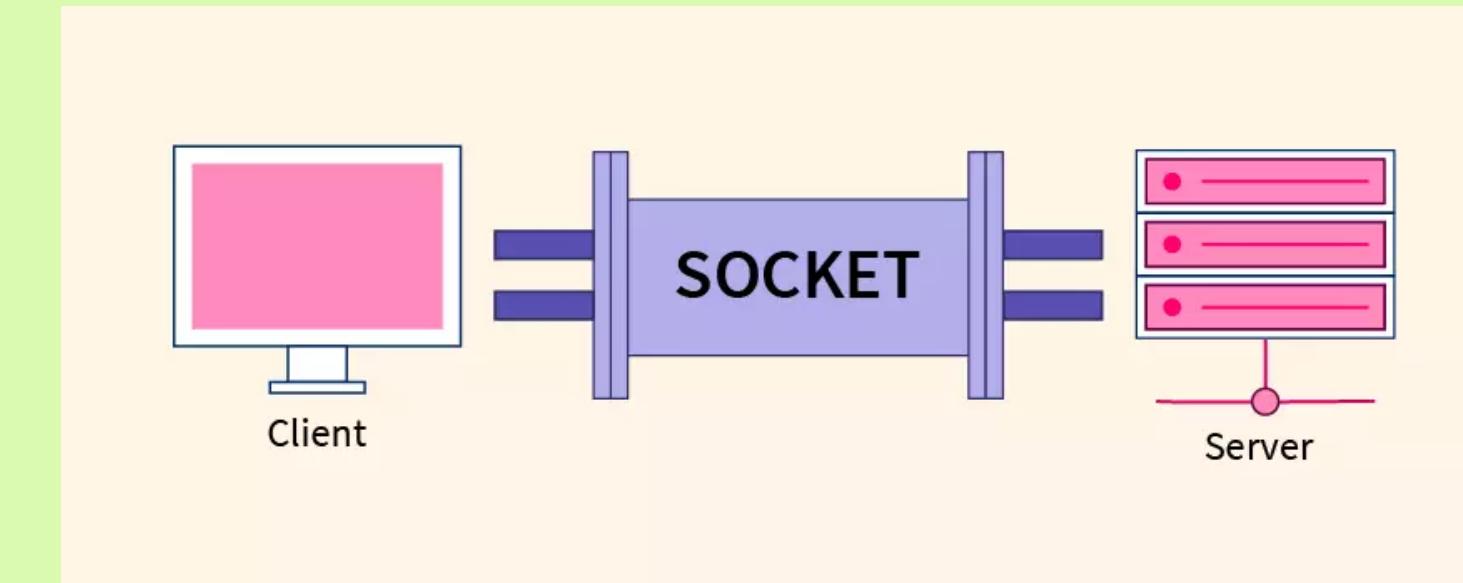


TKinter



PILLOW

SOCKET-PROGRAMMING



1

SOCKET

- Endpoints of a bidirectional communications channel
- Communicate within a process, between on the same machine or between processes on different continents

2

SOCKET-PROGRAMMING

- A computer networking concept
- A way of connecting two nodes on a network to communicate with each other over a network using sockets.

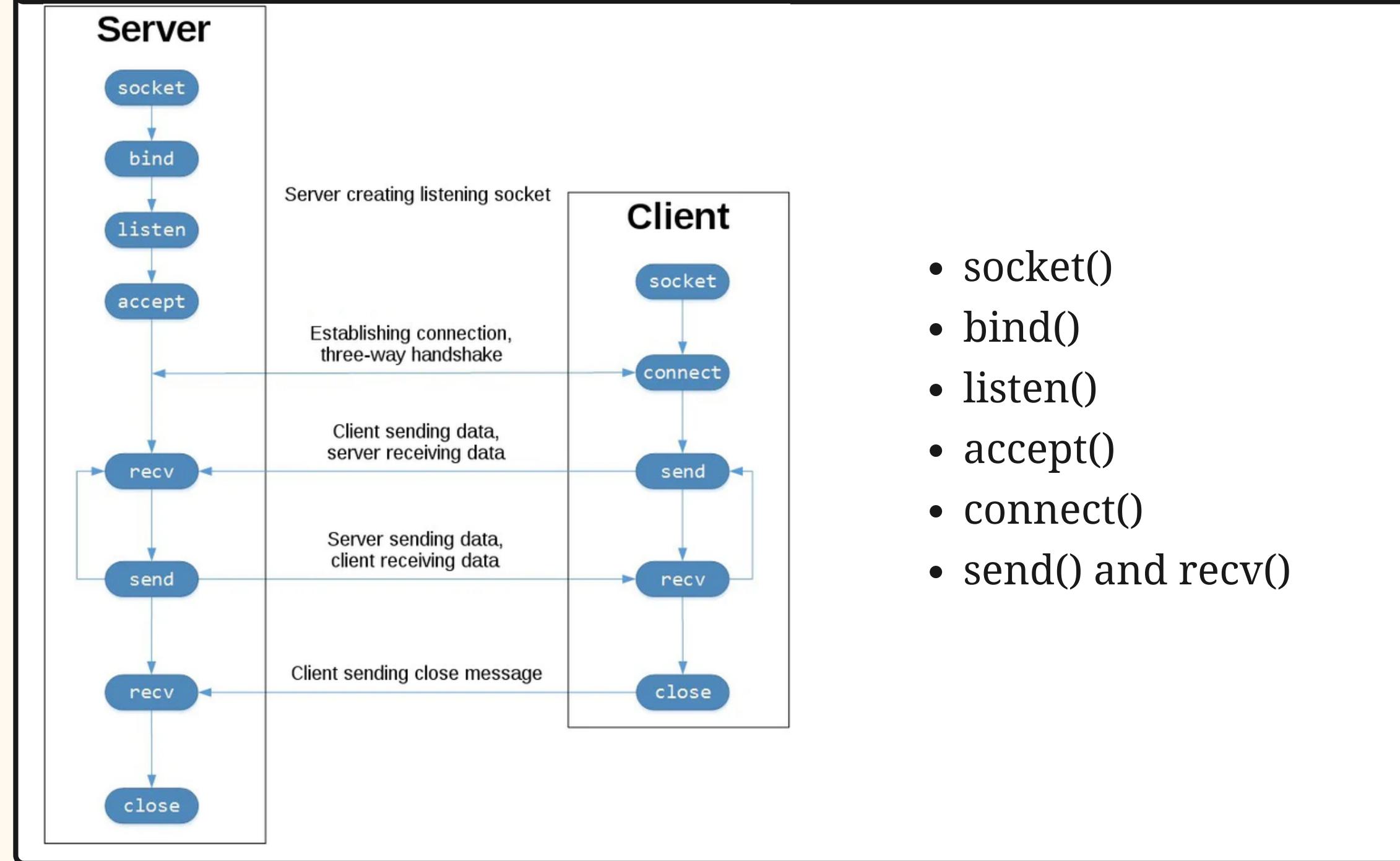
3

WHY???

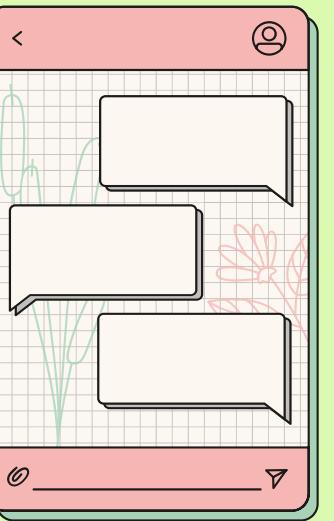
- Chat apps require real-time communication between clients and a central server.
- Sockets facilitate data exchange, allowing users to send and receive messages instantly.



KEY FUNCTIONS



TKINTER



1

WHAT IS TKINTER?

- Standard GUI library for Python.
- Provides a set of tools for building windows, dialogs, buttons, text fields, and other GUI elements.

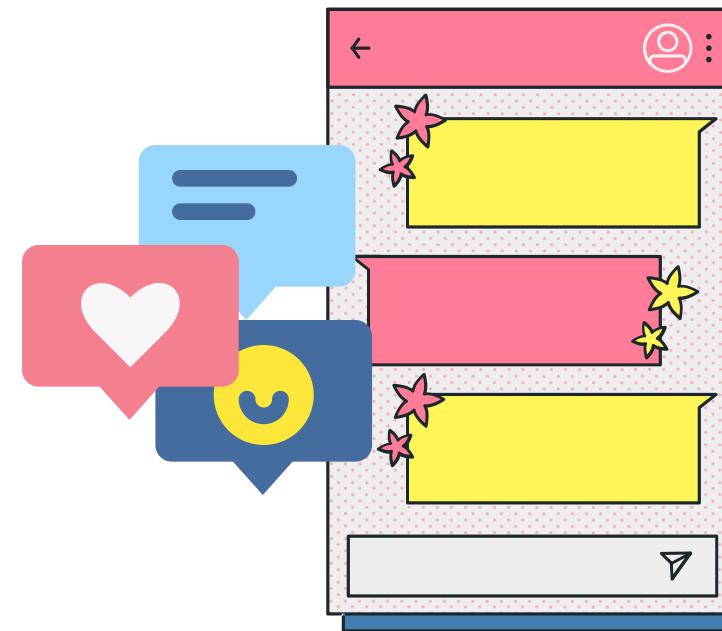
2

WHY???

- To create a user-friendly interface for users to interact with.
- To design and organize the visual components of your chat app.



KEY FEATURES



- Frame
- Labels
- Button
- Entry Fields
- Text Widgets
- MessageBox
- Filedialog Box
- Emoji Support

PILLOW



1

WHAT IS PIL?

- Python Imaging Library (PIL) to work with images in various formats.
- Provides functionalities for image processing, including opening, manipulating, and saving images.

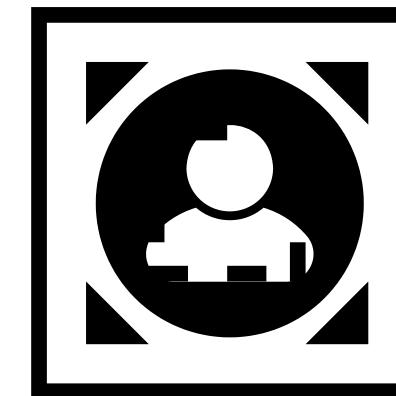
2

WHY???

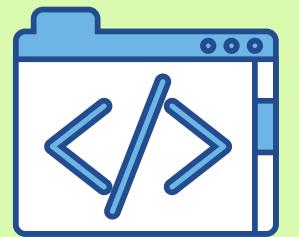
- used for various image-related features, such as sending and displaying images in messages.
- Helps in resizing, cropping, and enhancing images within the chat interface.



KEY FEATURES

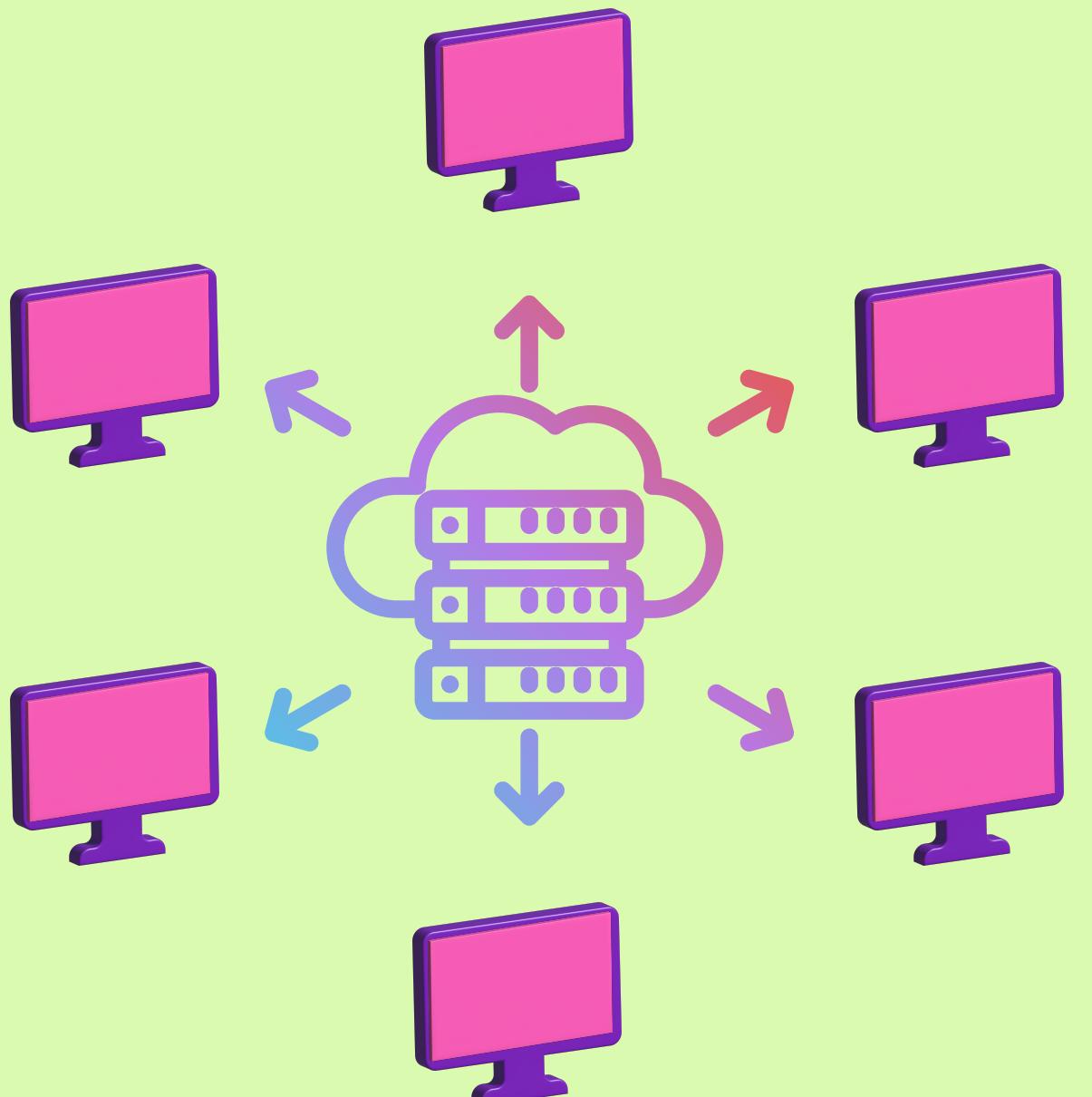


- Image Opening and Display
- Image Resizing
- Image File Format Support
- Image Saving
- ImageTk Integration
- Image Manipulation

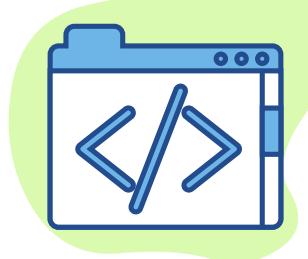


PROJECT ARCHITECTURE

ARCHITECTURE



INITIATION



The connection will be first initiated in the server side.

CONNECTING WITH IP



All the clients who are under the same IP address can be connected with the server.

DATA SAVING

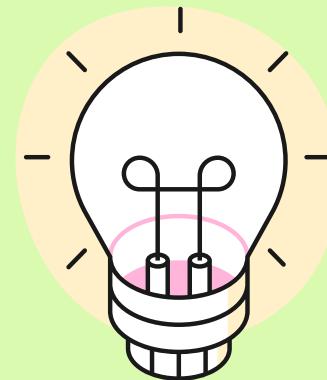


The Users will have to log in to the chat upon entering the system. All the user info will be transferred towards the server.

CHATTING



Upon multi-user entries, the conversations will be packed and forwarded from the server.



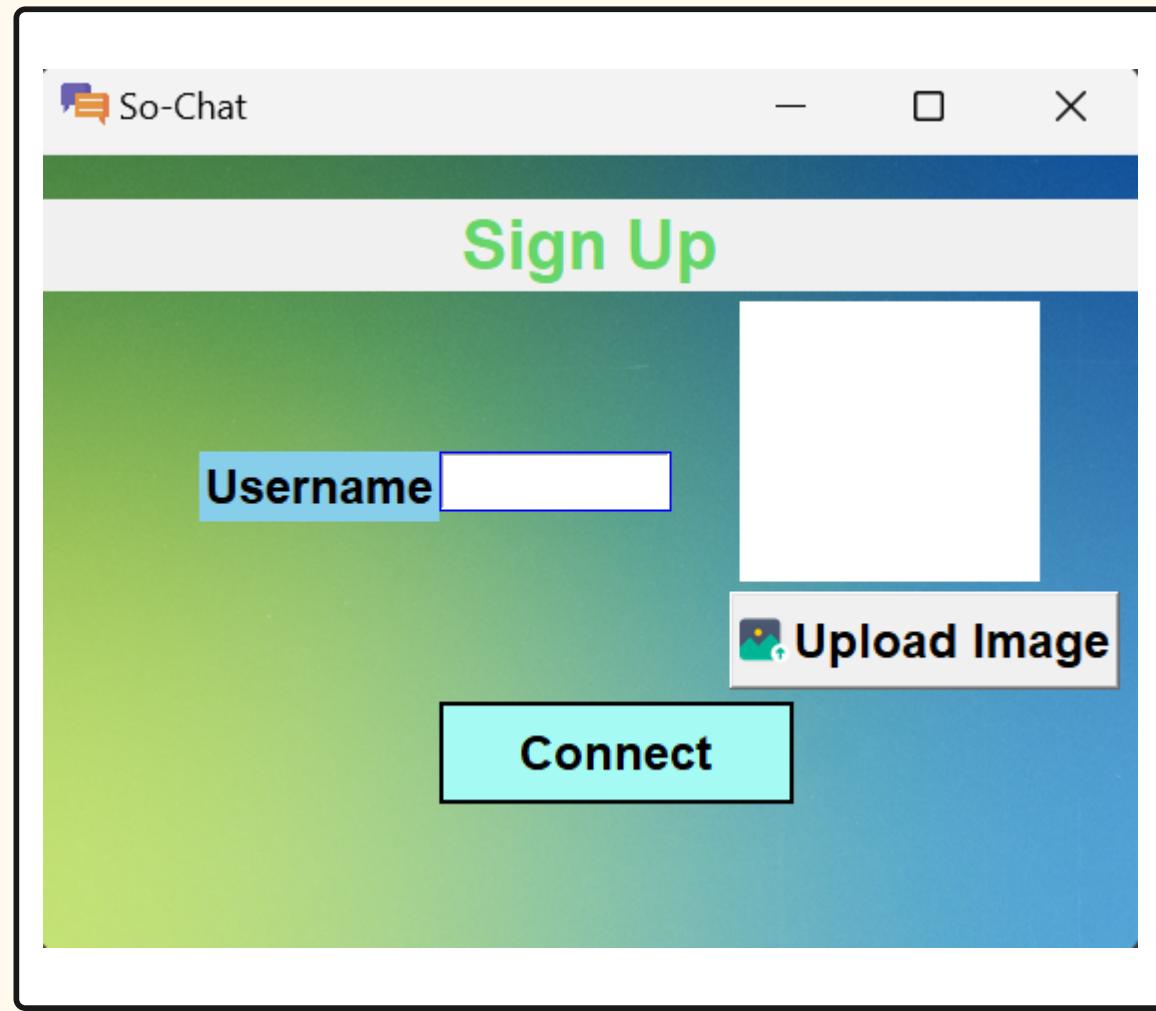
BENEFITS ON DECENTRALIZED COMMUNITIES

- **Real-Time Communication:** Users can engage in real-time conversations with members, making it convenient for organizing events or discussing local issues.
- **Information:** Users can stay informed about news, events, and developments that directly impact the organization.
- **Collaboration:** It is ideal for collaborative work and project management. Team members can discuss tasks and coordinate efforts in one centralized space, reducing the need for lengthy email threads or in-person meetings.
- **Project Tracking:** Team members can use decentralized chat to track project progress, discuss milestones, and provide updates on tasks, ensuring that everyone stays on the same page.
- **Reduced Email Volume:** Group chat can reduce the need for lengthy email exchanges, freeing up email inboxes for more formal communication and reducing clutter.

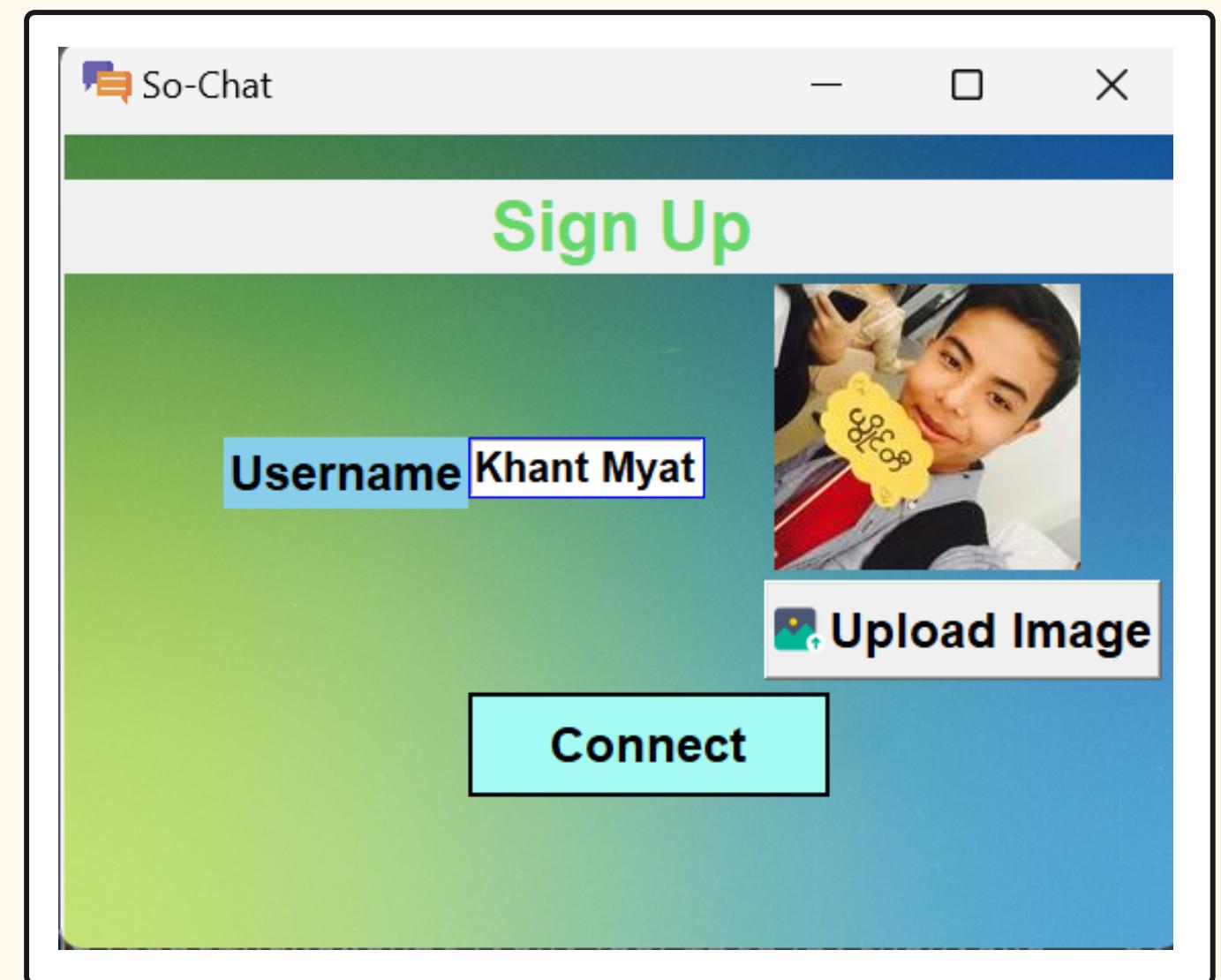


TESTING & RESULTS

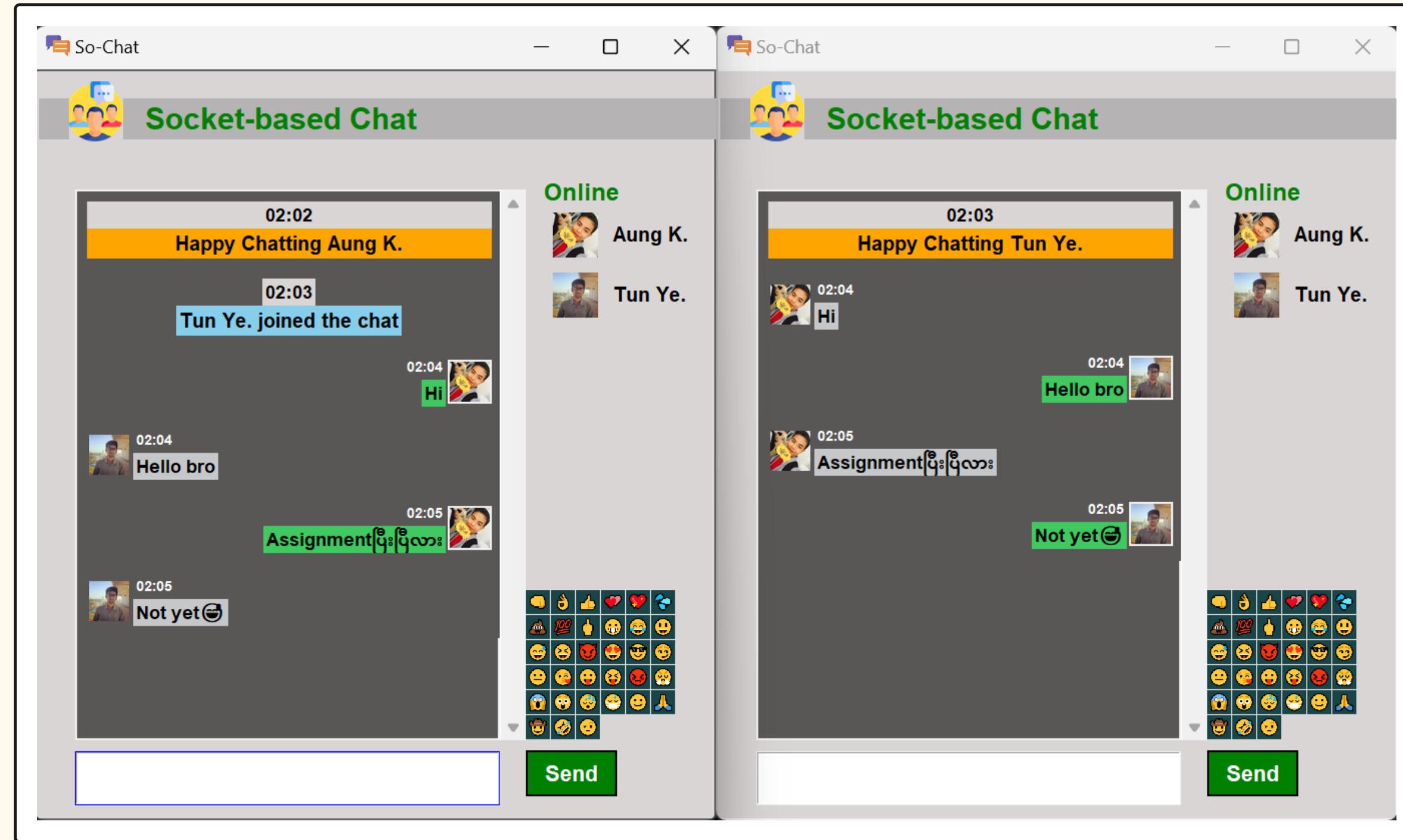
- server.py
- client.py



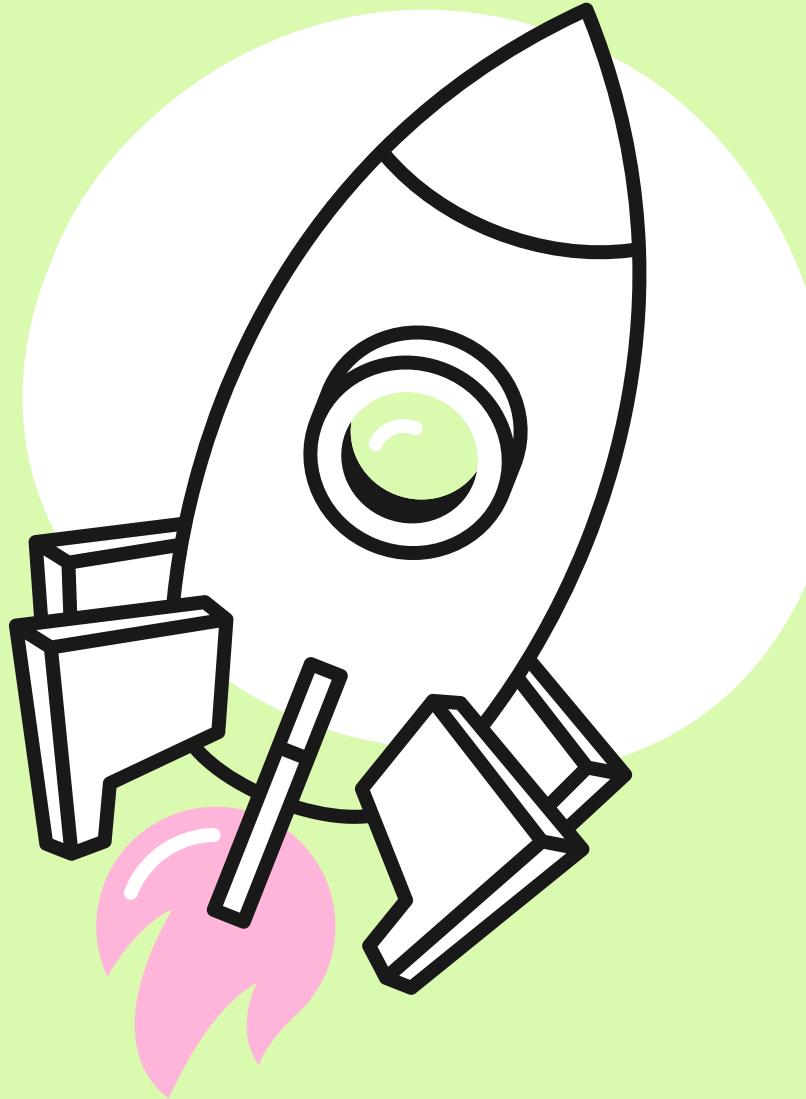
- Sign Up Interface



- Sign Up Interface



- Chat Screen Interface



**FUTURE
WORKS**

WHAT'S THE NEXT STEPS?

1

SECURITY

Strengthen security measures, including encryption to protect sensitive data and communications. This will protect us from unauthorized access if the server is compromised.

3

FILE SHARE

Allow users to share work or project-related files within the organization. Allowing all kinds of documents will boost the organization's work efficiency.

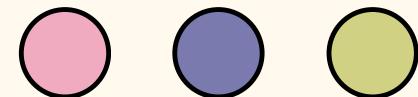
2

ARCHIVING AND SEARCHABILITY

4

SCALABILITY

Ensure that the chat application can scale effectively to accommodate a growing number of users and messages while maintaining performance and responsiveness.



SO-CHAT

**THANK
YOU**